EasyAlbum: An Interactive Photo Annotation System Based on Face Clustering and Re-ranking*

Jingyu Cui†, Fang Wen‡, Rong Xiao‡, Yuandong Tian§, Xiaoou Tang‡

†State Key Lab of Intelligent Technologies and Systems
Department of Automation
Tsinghua University, Beijing, P. R. China
cui-jy@mails.tsinghua.edu.cn

‡Visual Computing Group
Microsoft Research Asia, Beijing, P. R. China
{rxiao,fangwen,xitang}@microsoft.com

§Center for Brain-like Computing and Machine Intelligence
Shanghai Jiaotong University, Shanghai, P. R. China
tydsh@sjtu.edu.cn

ABSTRACT
Digital photo management is becoming indispensable for the explosively growing family photo albums due to the rapid popularization of digital cameras and mobile phone cameras. In an effective photo management system photo annotation is the most challenging task. In this paper, we develop several innovative interaction techniques for semi-automatic photo annotation. Compared with traditional annotation systems, our approach provides the following new features: "cluster annotation" puts similar faces or photos with similar scene together, and enables user label them in one operation; "contextual re-ranking" boosts the labeling productivity by guessing the user intention; "ad hoc annotation" allows user label photos while they are browsing or searching, and improves system performance progressively through learning propagation. Our results show that these technologies provide a more user friendly interface for the annotation of person name, location, and event, and thus substantially improve the annotation performance especially for a large photo album.

ACM Classification Keywords
H.5.2 User Interfaces: [Graphical user interfaces (GUI), Prototyping]; H.3.3 Information Search and Retrieval: [Clustering, Information filtering]

*This work was done when Jingyu Cui and Yuandong Tian were visiting students at the Microsoft Research Asia.

Author Keywords
photo tagging, face tagging, annotation, cluster annotation, face recognition

INTRODUCTION
Digital photo albums are growing explosively in both number and size due to the rapid popularization of digital cameras and mobile phone cameras in the last decade. Photos are easy to take with one-click shot and the cost of taking and storing a photo has been reduced to almost zero. As a result, the size of digital photo album is increasing exponentially. Automatic management of these large photo albums become indispensable.

Related Work
Traditional photo management systems only utilize the time or album/directory information which can be reliably extracted to help user manage the photos. However, these information are not sufficient to achieve good organization and search performance.

For efficient photo organization, a straightforward solution is to use semantic keywords for the photo management, such as: who is in the photo; where is it; when the photo is taken; what happens; and what kind of photo it is (portrait photo, group photo, or scenery photo). However, due to the technical difficulties in automatic recognition of face, scene, and event, existing methods cannot guarantee an accurate result if the annotation process is fully automatic.

Most of contemporary softwares do this manually. They allow users either label photos one by one, or manually SELECT the photos that are believed to have the same label, and apply label to them after selection. The latter approach is called *"batch annotation"* or *"bulk annotation"* [15], and is adopted widely in many commercial photo album management softwares, such as iView Media Pro [1], ACDSee [2], Microsoft Digital Image Suite [3], Picasa [4], and Photoshop

Elements [5]. This kind of approaches are often combined with drag and drop style user interfaces [14, 19] to further facilitate the annotation process.

Although these photo annotation user interfaces do reduce workload compared to annotate images one by one, users still have to manually select photos to put them in a batch before *"batch annotation"*. This is especially boring for face annotation, because before each annotation, users need to recognize all the people in all the candidate photos and manually verify that the selected ones do contain a certain person. According to the study in [18], people have less motivation to invest the effort in annotating their photos due to heavy workload of manual annotation. Mostly they will store their photos in hard disk without annotation. Therefore, how to lower the labor cost barrier for manual annotation is still a major challenge.

There are also some existing methods trying to leverage the potential of Internet to alleviate the pain of tagging photos, by means of encouraging the Web users to label the photos online (Flickr [6]), or implicitly labeling the keywords of photos in games (research paper [22], ESP Game [7], Google Image Labeler [8]).

However, these Internet based annotation methods are not satisfactory enough because they also require extensive labeling and tagging workload. Moreover, people are not always willing (or capable) to label photos of others, and some people are not willing to share their family albums to public.

The idea of clustering is recently introduced trying to do the "selection" automatically. Time based clustering is quite easy, and is used in softwares such as Microsoft Digital Image Suite [3] and Picasa [4]. In [16], Platt proposes to use content-based feature to cluster photos automatically. However, according to his later study in [17], this kind of feature is less robust than time feature extracted from photo meta-data. Time based clustering does help in organizing and browsing photos, but contributes little in annotating people names in photos since it cannot cluster photos with the same people together.

In recent years, remarkable progress has been made in computer vision. Especially, the performance of automatic face detection method has improved significantly [21]. Photoshop Elements [5] detects faces in photos and allows user manually label these faces directly. However, this method does not do any clustering, thus till requires a great deal of labor similar to annotation directly on photos. Zhang et. al. [25] uses photo context information to generate a candidate name list for each detected face. For both methods, each face still need to be annotated one by one.

Bongwon et. al. [20] extracts torso color based on face detection results, and do clustering based on torso clothes color. However, it can only cluster photos from the same day. Even for the same day, if a person puts on/takes off a jacket, or two persons wear similar colored clothes, the approach fails. This can surely happen for group photos given the limited number of popular colors, e.g for mans cloth. Moreover, detecting cloth color reliably is itself a challenging topic. No experimental results are provided in their study.

Recently, Automatic Face Recognition techniques have also been developed with increasing efficiency and performance [23,24,27]. However, in practical applications, face recognition is only used for similar face ranking [9,13,26]. Given the diversity of family photo (indoor, outdoor, with/without eyeglasses), current face recognition algorithm performs poorly for ranking all faces this way. For annotation application, recall rate is very low. In addition, users still need to click on EVERY single photo to annotate, requiring the same large number of user interactions as conventional methods.

Our Method

In this paper, we propose a novel user interface for semi-automatic photo annotation. We address existing problems of photo annotation through a novel combination of face recognition technology and interactive UI. The key components of the system include:

1. Cluster annotation

   Our system, *EasyAlbum*, introduces a novel annotation UI called cluster annotation which annotates photos by clusters instead of one by one. The new UI consists of two parts. One is automatic clustering, which groups photos/faces with similar scene/facial feature into clusters. Another is a set of pre-defined interactive annotation operations, which enables user to conduct photo/face annotation on clusters directly. With this UI design, the annotation workload is dramatically reduced.

2. Contextual re-ranking

   EasyAlbum provides a context-sensitive environment adapting to the user's interactions. This helps to relieve the burden of users. During the interaction of the user and the system, the operations of the user actually reflect the intention of the user. We can guess this intention and re-rank the photos or clusters to be labeled. For example, when the user clicks one photo/cluster, it indicates his/her attention on this one. We re-rank the photos/clusters and arrange the similar ones close to the clicked photo/cluster. The user will find it easier to label them together. This feature boosts labeling productivity of users.

3. Ad hoc annotation

   In our system users are allowed to annotate photos in an ad hoc manner when they are browsing or searching. The annotation information is accumulated gradually and then used to progressively improve the clustering and search results of unlabeled data. Unlike existing system in which labeled data do not affect unlabeled data (static), this dynamic interaction between labeled and unlabeled data can significantly improve user experience since users can see improved browsing and retrieval results as they annotation.

INTERACTIVE ANNOTATION

We provide a novel User Interface (UI) to label people's name, as well as location and event. This UI is designed to work in a semi-automatic interactive way. We facilitate the interactions between users and the system in all circumstances, and reduce the workload of labeling as much as possible.

Our system can be characterized by the following new features: cluster annotation, contextual re-ranking, and ad hoc labeling. Since location/event annotation has a similar strategy with face annotation, in the following description we will mainly focus on the face annotation and then give an introduction of location/event annotation in a subsection.

Cluster annotation

Traditional tagging methods involve workload proportional to the size of the album when labeling scenes or faces, because the user has to label the photos individually, or confirm selections one by one before labeling in a "batch". Actually, these albums are often redundant in nature, i.e. for one photo, there are often a group of photos similar to it, either in similar scene or have the same faces. Making full use of this redundancy, our system attempts to label groups of similar photos/faces in much fewer operations.

Taking face cluster annotation for example, after photos are imported to our system, a face clustering algorithm is automatically started. Technical details of this clustering algorithm is described in Section *Algorithm*. This algorithm groups faces with similar appearances.

Due to the limitation of clustering and face recognition algorithm currently available, a face cluster cannot be directly mapped to a person name without user interaction. The faces of the same person may be scattered in several clusters, also occasionally, different persons may be found in the same cluster. Therefore, face annotation can be carried out through a set of operations on clusters, such as cluster annotation, cluster merging and splitting. These operations enable user directly manipulate a group of photos instead of annotating them individually.

UI of cluster annotation

We design a Windows Explorer style UI to allow users to manipulate *FaceGroups* in the similar way of manipulating folders and files in Window Explorer (Figure 1). Those who are familiar with Microsoft Windows System will find it rather easy to use our system without any training.

As presented in Figure 1, the *Group View Area* presents the thumbnails of *FaceGroups* to the user, giving a preview of the faces contained in each *FaceGroup* and allows the user to label confident *FaceGroups* directly. The *Thumbnail Area* dynamically shows the faces of currently selected *FaceGroup*, allow users to observe the faces in a *FaceGroup* without double clicking and diving into the *FaceGroup*. In this UI, three operations are defined for face annotation, including cluster/single-face annotation, drag and drop, annotation deletion.

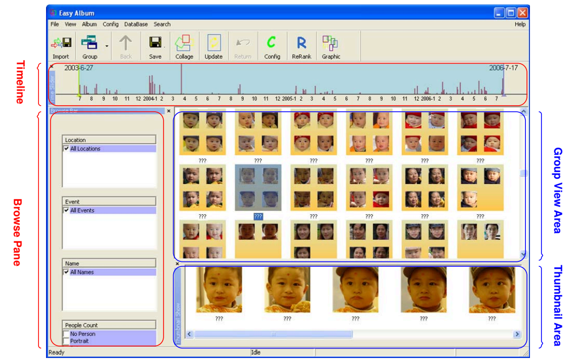


Figure 1. The *EasyAlbum* Face Labeling User Interface

Cluster annotation is supported in the context menu when user right clicks on a thumbnail, suggesting the existing name tags in the database. The suggested name tags are ranked according to the similarity between faces in the selected cluster and the faces of the tagged person. The user is allowed to select a name from the list, or type a new one. This further accelerates the speed of annotation. It even allows annotation only with drag and drop after several names already exist in the database. Figure 2 shows a demonstration of clicking on a suggestion to label the *FaceGroup* as "Tom" (left) and typing a new name to label another *FaceGroup* as "Rose".

Figure 2. Annotation by selecting an existing name or type a new one

When users want to correct some of incorrectly labeled faces in a *FaceGroup*, they can use single face annotation as shown in Figure 3, or simply remove the incorrect faces from the cluster through drag and drop described below.

The UI also supports drag and drop operation similar to the operations in Windows Explorer. User can drag faces or clusters without annotation to an annotated cluster, which means assign an annotated name to these faces or clusters.

Here are some examples. Figure 4 demonstrates dragging a face to the *FaceGroup* "Tom". The face is automatically tagged to the name "Tom". Figure 5 shows a demonstration of dragging a *FaceGroup* to the *FaceGroup* "Kate". All the faces in the dragged *FaceGroups* is tagged to the name "Kate".

When users want to delete the name tags of faces that are already labeled, they can simply press DELETE on key board

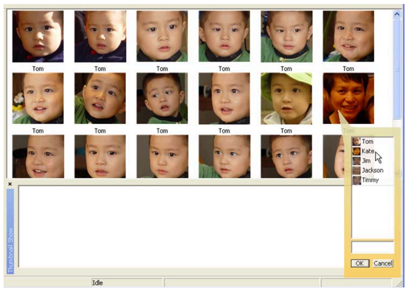


Figure 3. Annotation of a single face. There is one mistake in group "Tom", this operation will correct its name as "Kate"

Figure 4. Drag from Thumbnail Area to Group View Area

or drag the face or *FaceGroup* to an "Unname" folder (Similar to Recycle Bin in Windows), the name of the selected faces will be automatically reset.

EasyAlbum Explorer Style UI gives a friendly and efficient way of labeling photos. Especially, the ability to cluster similar faces/scenes together is a distinctive feature compared with contemporary softwares. Through drag-and-drop operations, as well as keyboard interactions, a user can accomplish the labeling job much more efficiently than using traditional methods.

Smart Cluster Merging and Splitting

The state-of-the-art cluster algorithms, including our method, cannot guarantee perfection of the automatic clustering result. Thus, after clustering, a *FaceGroup* may contain faces of more than one person. *EasyAlbum* handles this effectively by providing a flexible interactive mechanism to allow users

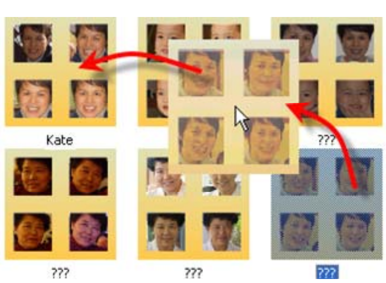
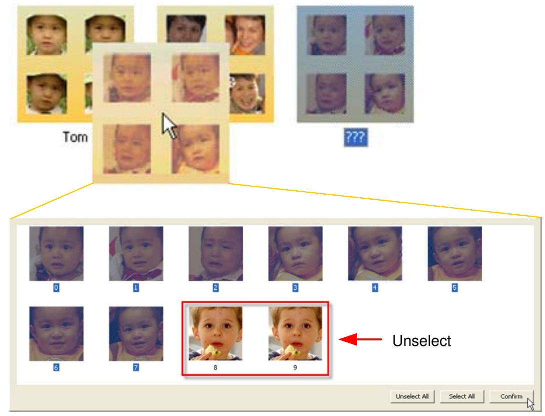
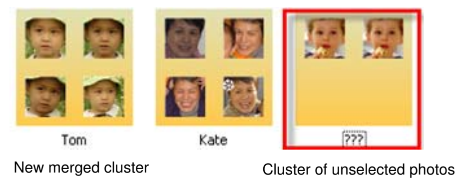


Figure 5. Drag between FaceGroups in Group View Area



(a) After dragging, unselect misclustered photos before merging with labeled clusters



(b) After merging, two clusters are generated: the merged cluster and the cluster containing unselected photos

Figure 6. Smart Cluster Merging

to interfere in the process of labeling and correct the mistakes the algorithms have made.

When users drag several faces or *FaceGroups* to a *FaceGroup* that already has a name, this indicates the user wants to name all of the dragged items together. However, the dragged faces may not all belong to the same person, due to the imperfection of the algorithm currently available, and dropping these faces into an already well annotated *FaceGroup* will spoil the work done before. To solve this problem, we will arrange the dragged faces according to their similarities to the destined *FaceGroup*, and allow users to select part of the dragged face group to merge (or unselect unwanted ones). The unselected faces will just be left alone and their labels will not change at all, as shown in Figure 6. This merging process guarantees that the faces will be assigned to the right labels as the user specifies.

Also, after Re-Ranking in the *FaceGroup* process, the faces that are less similar to most of the faces in current *FaceGroup* will be arranged together (See Figure 7). If they actually do not belong to the current *FaceGroup*, the user can easily select them and simply press DELETE on keyboard, the original *FaceGroup* will be automatically split into two, one contains the faces left in the original *FaceGroup*, the other holds the deleted faces, enabling further labeling.

Contextual re-ranking

EasyAlbum will record the contextual information when the users are clicking, dragging, and dropping the faces or *FaceGroups*, either in browsing mode or annotating mode, and re-rank the thumbnails in current view to enable users to

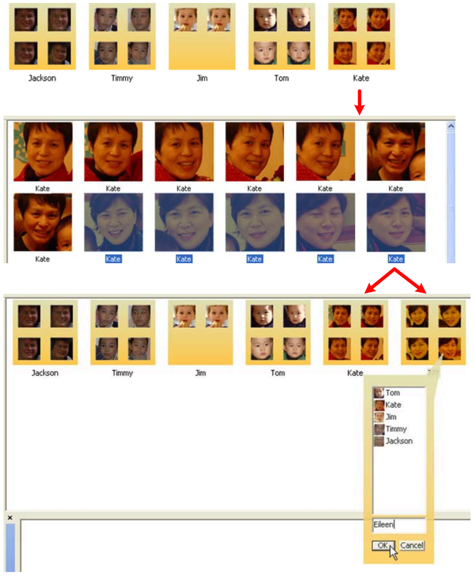


Figure 7. Smart Cluster Splitting. One of the original clusters contains faces of two similar people, we can select out the faces of another person (grayed ones in the middle part), press DELETE on keyboard, and smart splitting will be performed to enable further labeling (bottom).

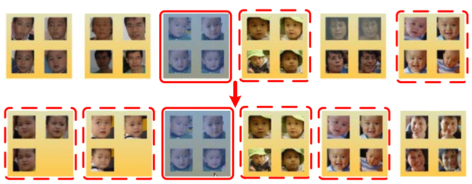


Figure 8. Arrangement of FaceGroups before and after Re-Ranking. Note that before re-ranking, there are only two FaceGroups belong to the same subject of the selected one in the neighborhood, while after re-ranking, there are four, and all are positioned near the selected one.

accomplish the interactive labeling job without much scrolling the *Thumbnail Area*.

For instance, when users are browsing the *FaceGroups* and clicking on one *FaceGroup*, *EasyAlbum* will automatically adjust the order of the *FaceGroups* in current view to put the similar *FaceGroups* near the *FaceGroup* that is clicked. This will make the merging of several *FaceGroups* of one person more efficient. Without this feature, the user might need to scroll a long way to find two *FaceGroups* to merge.

Figure 8 gives a comparison of the order of *FaceGroups* before and after Context Based Re-Ranking is carried out. It is clear to notice that after re-ranking, there are more similar *FaceGroups* near the selected one so that further merging becomes more convenient.

When users browse in a *FaceGroup*, *EasyAlbum* will also arrange the face thumbnails in it to put the faces similar to the selected one together, and give convenience to further manipulation, such as dragging several similar faces together, or deleting from current *FaceGroup* the images which are mis-

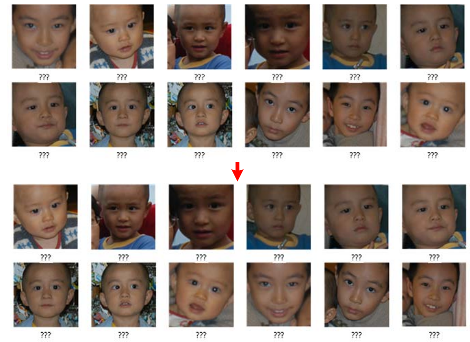


Figure 9. Comparison of orders of Faces before (top) and after (bottom) Re-Ranking

specified to this *FaceGroup*. Figure 9 shows an example of such re-ranking.

Ad hoc annotation with progressive learning

We introduce ad hoc annotation to make use of user labeled data for learning so to improve clustering of the unlabeled data progressively. As the user casually annotates photos while browsing and retrieving photos, the annotation information is accumulated gradually, and it in return continuously improves the clustering, browsing, and retrieval performance. Unlike previous system in which labeled data do not affect unlabeled data (static), our system improves its performance dynamically. This feature is important for user to label photos freely at a casual pace. It turns a tedious job into a fun game. As the user sees the improved browsing and retrieval results, he/she is then further encouraged to annotate more photos/clusters.

When the photo album to be labeled is huge and contains many subjects, ad hoc annotation allows filtering as a preface to clustering to improve automatic clustering results. The users can label the huge album hierarchically by first filtering the album by some selection criteria (such as time span), then carrying out clustering on the selected photos, which is manageable in size.

Moreover, ad hoc annotation also allows users annotate as they browse and search (detailed browse and search features are described in the next section). Since our system extracts image appearance based feature, browsing and searching can be started based on theses feature before any manual annotation information are input. User can then label on search results, or enjoy improved search based on labeling.

Semi-Automatic Location and Event Annotation

Similar to person name annotation, we also enable annotation of location and event, as shown in Figure 10. The only difference between face annotation and location/event annotation is that we use different features to do the automatic clustering.

In the location and event annotation, unlike some of the existing methods which make use of only the time and GPS in-

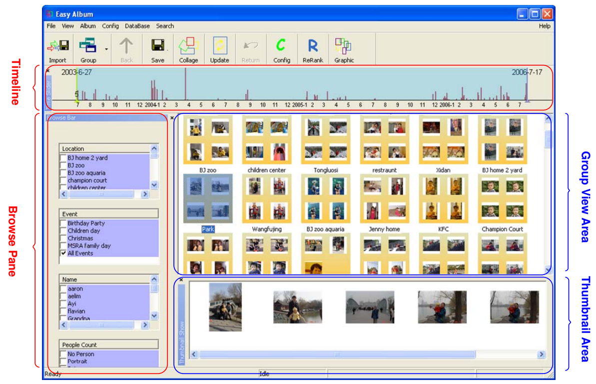


Figure 10. The *EasyAlbum* Scene Labeling User Interface

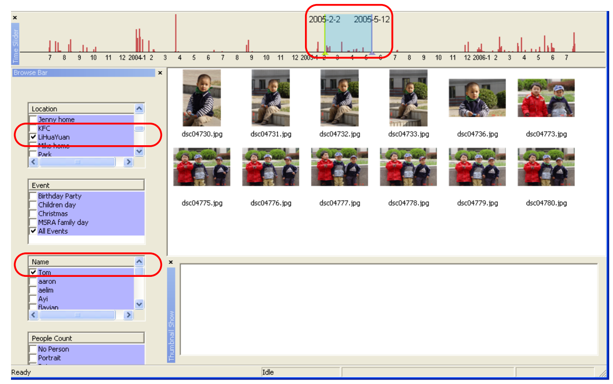


Figure 11. Demo of Browsing in *EasyAlbum*

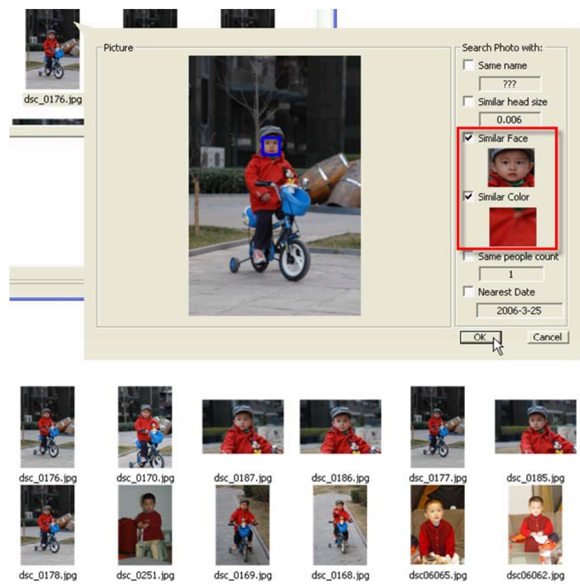


Figure 12. Demo of Searching in *EasyAlbum*

formation in JPEG tag, we take into consideration the analysis of photo appearance, including color model and texture.

The feature of semi-automatic location and event annotation makes it possible to recognize different locations in similar dates, which is especially useful when the photo album consists of photos taken by different people at the same date, or the photographer is traveling and take pictures of different places in one single day.

BROWSING AND SEARCHING: BASED ON INTERACTIVE ANNOTATION

The interactive annotation mechanisms mentioned above help us to get name, event, and location tags of photos efficiently. With the obtained labels, we can easily browse and search photos with *EasyAlbum*.

EasyAlbum enables 5-Dimensional Photo Exploration, which means browsing in the photo albums along five dimensions or their combinations. We design a timeline (See Figure 1) to give the user a brief view of the distribution of photos with respect to time. The user can specify an interval and view the photos within it. The other four dimensions include: Location, Event, Member of the photo (Name), and People Count in the photo. All of them are integrated in Browse Pane (See Figure 1). Figure 11 shows the result of browsing the photos satisfying the condition: between 2005-2-2 and 2005-5-12, at LiHuaYuan, and with Tom in the photo.

In addition, *EasyAlbum* provides a feature of searching from a single photo. Since we have got both photo feature and semantic tag information from preprocessing and interactive labeling stages, semantic search (such as search photos that contain some specified people) and feature similarity search (such as search by face similarity) are made easy. Figure 12 shows the UI of right-clicking on a photo and select search by face and cloth similarity (top), the results obtained are also shown (bottom). It is satisfying to see that the photos retrieved all contain the same person (Tom) wearing cloths with similar color. Notice that in this search example we do not even need prior tagging information. So we do not have to wait for annotation to use search feature. Thus search and annotation can be iteratively conducted in our system.

ALGORITHM

EasyAlbum is developed to help user annotate photos semi-automatically by leveraging the state-of-the-art computer vision technologies. This system uses a three-stage architecture: offline preprocessing, online clustering, and online interactive labeling.

In the offline stage, three sets of features are extracted: 1. Scene feature, a Color Correlogram feature [12] which is widely used in CBIR applications to describe the color and texture information; 2. Time feature, can be extracted from the meta-data of image files directly; 3. Facial features, can be extracted after face detection [21] and alignment procedure [28,29]. There are two kinds of facial features: a. Local binary patten (LBP) feature [10], a widely used feature for face recognition; b. Cloth contexture feature, Color Correlogram feature extracted from the human body area.

In the online clustering stage, an enhanced version of spectral clustering algorithm [11] which can handel noise data is

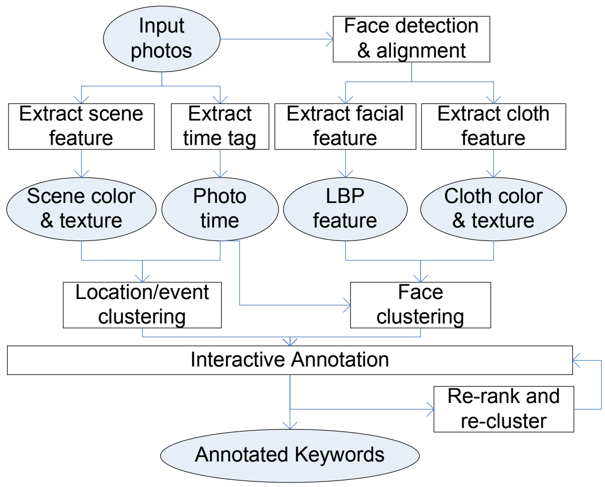


Figure 13. System framework

applied on these three sets of features, as shown in Figure 13, to group similar faces into clusters, or group similar photos with similar scenes. For face clustering, we allow the system to produce multiple clusters for a single person to address the face diversity. Clusters of the same person can then be manually combined by a few drag and drop operations.

In the interactive stage, two kinds of ranking technologies are used, called Intra-cluster Ranking and Inter-cluster Ranking. The Intra-cluster Ranking reorders the faces in each cluster so that more confident faces are put in front. Inter-cluster Ranking rearranges clusters so that better ones are put in front and 'similar' clusters are put together to reduce user operations.

Both re-ranking technologies dynamically change spatial configuration of photos/clusters according to newly-input ground-truth from users. These methods can be implemented by supervised learning method, including linear discriminant analysis [23, 24] or support vector machine (SVM), or simple nearest-neighbor. We use simple and efficient nearest-neighbor approach for re-ranking, because in most cases, simple method tends to produce more stable results with a smaller number of training samples.

EXPERIMENT AND USABILITY TESTING

We conduct an user study to compare *EasyAlbum* with *Adobe Photoshop Elements 4.0*, which is representative of existing album management softwares including ACDSee Pro 3.0 and iView Media Pro 3.1.

All of the softwares above provide features of creating tags, dragging and dropping photos and exploring according to tags. As a special feature, *Adobe Photoshop Elements 4.0* can automatically detect faces in photos and allow users to directly label the faces instead of photos. This makes *Adobe Photoshop Elements 4.0* superior to ACDSee Pro 3.0 and iView Media Pro 3.1. As a result, we decide to compare *EasyAlbum* to *Adobe Photoshop Elements 4.0* to demonstrate the performance of *EasyAlbum*.

Experiment Design

Our experiment consists of two parts: 1. Test the scalability of *EasyAlbum* v.s. *Adobe Photoshop Elements 4.0*; 2. Compare the detail performance when the album size is relatively small.

We recruit 32 volunteers from university campus to take part in our user study. Their grades vary from freshman to graduate grade 3. Their ages range between 19 and 24.

According to our questionnaire, among the 32 volunteers, 97% (31) believe that it is necessary to organize the personal album by "Who is in the photo", 97% (31) of them are not willing to share all of their photos to public (such as upload to website) to let others label for them.

Before the test begins, each subject is given a training on how to use *EasyAlbum* and *Adobe Photoshop Elements 4.0* to tag the names of faces in an album. The operation method of each software is tutored thoroughly. All attendees are encouraged to practice on both softwares until they are fully familiar with both systems.

We collect a large personal album with ground-truth and import it into our database. The albums used in the following experiments are drawn randomly from it. Before each of the experiments, representative photos containing the individuals in the database are shown to the attendees to let them recognize the people in the album.

We define the labeled rate $p1$ as the proportion of labeled faces to all the faces. We also calculate the accuracy $p2$ as the proportion of correctly labeled faces to the labeled faces. We take harmonic average of these two criteria to measure the performance of a single annotation, i.e., $p = 2/(1/p1 + 1/p2)$, to balance both the amount of faces labeled as well as the accuracy of labeling. This is similar to the common method in the field of Information Retrieval which uses harmonic average of recall rate and precision rate to measure the performance of a retrieval algorithm.

Scalability Comparison

To test the scalability of the two labeling softwares, we randomly select 15, 50, 100, 200, and 400 photos from our database and invite 5 of the volunteers to label them using both *Adobe Photoshop Elements 4.0* and *EasyAlbum*. The number of subjects in the album is kept constant as 5, to simulate the case of ordinary family album. To eliminate possible ordering effect, we alternate the order of softwares for different users.

The averaged time consumption of the labeling job using the two softwares is shown in Figure 14. It can be observed that the time needed by *Adobe Photoshop Elements 4.0* grows approximately linearly with the size of the album, while the time of *EasyAlbum* grows sub-linearly (remains almost constant for large size). Our system is much superior for annotation of a large album. For example, in Figure 14, compared with Photoshop Elements, our system only uses less than

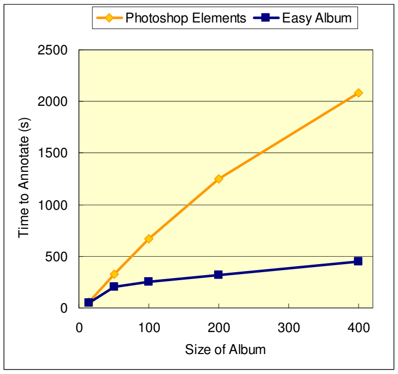


Figure 14. Scalability of *EasyAlbum* and *Adobe Photoshop Elements 4.0*

25% percent of time for annotation when the album contains 400 photos.

These results show that using cluster annotation, a large proportion of faces can be annotated in the early stage by a few interactions. The unlabeled faces in the "Unname" folder are small in number. In contrast to the one-by-one annotation UI style used in Photoshop Elements, our system shows significant performance improvement when annotating large albums.

Nowadays, digital camera owners often take hundreds of photos in a single day trip, which makes large family albums common. A typical photo album may contain thousands of photos with some reaching tens of thousands. *EasyAlbum* shows great potential in these scenarios, while *Adobe Photoshop Elements 4.0* requires linear time consumption, which is prohibitive for large albums.

Now that when the size of the album is large, *EasyAlbum* clearly outperforms *Adobe Photoshop Elements 4.0*, we are going to compare the performance in case of small albums in detail to further demonstrate the advantage of our system. The small data size is also easier for us to conduct more experiments in detail with more users.

Performance Comparison in Small Albums

To test the performance of *EasyAlbum* with small albums, we use two albums randomly selected from our database. Each of the album contains 50 photos and 5 individuals. The first album contain 115 faces, and the other contains 104.

The users are asked to finish two tasks using each of the two softwares. For each software, there are two tasks. For the first task, users try to label all of the faces, and we measure the time and accuracy. For the second task, given a 2 minutes time limit, we count how many faces are correctly labeled.

We design 8 schemes for our experiments, involving different order of albums, softwares and requirements to the users (Time limit or Accuracy). See Table 1 for detailed explanation. We alternatively select one of the schemes for each of the users in case there is an ordering effect. Since there are 32 users, each of the schemes is done 4 times.

During the process of labeling, the activities of mouse and keyboard, including number of clicks of mouse, number of strikes of keyboard, and total distance of mouse cursor movement are recorded. These statistics are used to indicate the effort needed to annotate the album. We also record time consumption and final annotation result, to further evaluate the annotation performance of each test.

Annotation workload comparisons

We compare the annotation workload by comparing the total mouse cursor movement distance and time consumption of labeling the whole album.

After our experiment, many users report that the most boring part of labeling photos is the repeated drag-and-drop operation, while the clicking of mouse is less concerned. As a result, we measure the distance of mouse movement of *EasyAlbum* and *Adobe Photoshop Elements 4.0* when doing the same labeling job to compare the usability of the two softwares. The experiments show that when using *EasyAlbum*, the distance of the mouse movement is in average 70% of the distance of *Adobe Photoshop Elements 4.0* (Figure 15(a)). Several users report subjectively that one of the most distinctive improvements of *EasyAlbum* is enabling drag and drop between *FaceGroups* nearby, while *Adobe Photoshop Elements 4.0* needs to drag photos or faces to the side pane in the UI, which is quite a distance away. The repeated operation of long distance mouse move will reduce the user experience, especially when the album is large and labeling job becomes boring.

EasyAlbum also needs less time to finish labeling. In Figure 15(b), it is shown that for all schemes, the averaged time consumption over all schemes is reduced by about 30-80s when using *EasyAlbum*. Considering the total time need is about 5 min, this is an improvement of about 20%. This is consistent with result shown in Figure 14 (The second point on the plot). As the size of the album increases, the improvement can reach over four times.

Performance comparisons

There are two kinds tasks in our experiment: 1. time cost comparisons while annotating all photos; 2. performance comparisons while doing annotation in a time limit.

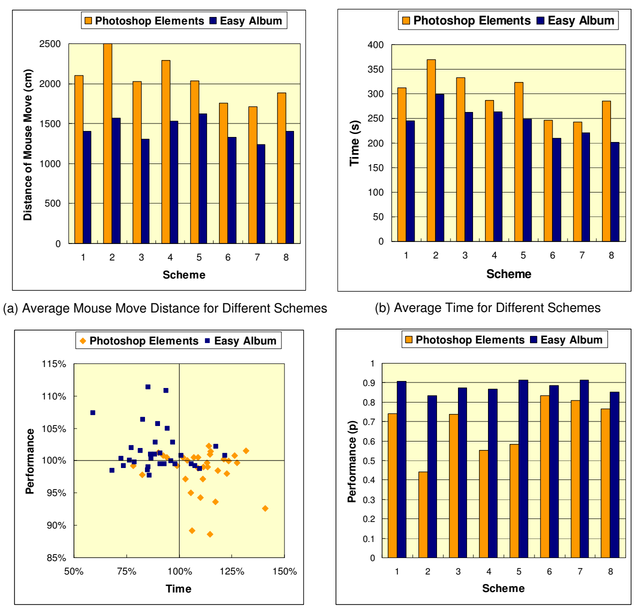
We analyze the result of the first kind of task by plotting performance to the time of each subject using the two softwares. The time and performance axes are normalized by the mean of each subject so that we can eliminate the effect of user's skill of operating the computer and align points of different users together to compare. Figure 15(c) shows that points of *EasyAlbum* are clustered at the upper left corner, indicating higher performance in less time.

As for the time-restricted task, *EasyAlbum* is a clear winner. Figure 15(d) shows a comparison of the final performance when 2mins run out. The result is averaged over all subjects in each scheme. *EasyAlbum* enables users to get an intermediate labeling result, by first automatic clustering and then user interaction, while *Adobe Photoshop Elements 4.0* only

Scheme	Album	Task1	Task2	Task3	Task4
1	1	EA, Accuracy	EA, Time Limit	PS, Accuracy	PS, Time Limit
2	1	EA, Time Limit	EA, Accuracy	PS, Time Limit	PS, Accuracy
3	1	PS, Accuracy	PS, Time Limit	EA, Accuracy	EA, Time Limit
4	1	PS, Time Limit	PS, Accuracy	EA, Time Limit	EA, Accuracy
5	2	EA, Accuracy	EA, Time Limit	PS, Accuracy	PS, Time Limit
6	2	EA, Time Limit	EA, Accuracy	PS, Time Limit	PS, Accuracy
7	2	PS, Accuracy	PS, Time Limit	EA, Accuracy	EA, Time Limit
8	2	PS, Time Limit	PS, Accuracy	EA, Time Limit	EA, Accuracy

Table 1. Different Schemes for User Study. EA stands for *EasyAlbum* and PS indicates *Adobe Photoshop Elements 4.0*

allows labeling faces one by one. When the time limit ran out, the users may be left with many photos unlabeled.



(a) Average Mouse Move Distance for Different Schemes (b) Average Time for Different Schemes

(c) Performance to Time of Different Users (d) Average Performance of Time-Restricted Task

Figure 15. Comparison: *EasyAlbum* v.s. *Adobe Photoshop Elements 4.0*

In summary, *EasyAlbum* outperforms *Adobe Photoshop Elements 4.0* in cases of both large scale and small scale. For very large albums, *EasyAlbum* provides a solution in tolerable time limit, while labeling by *Adobe Photoshop Elements 4.0* is tedious and boring for albums larger than 400. For small albums, *EasyAlbum* needs shorter mouse movement, and yields higher performance in less time compared to *Adobe Photoshop Elements 4.0*.

User feedback on the system

We send out questionnaires to all of the 32 users, and all of them responded in 24 hours.

Overall, subjects prefer *EasyAlbum* to *Adobe Photoshop Elements 4.0* to label the album. According to the result of the questionnaire, 91%(29) of them believe *EasyAlbum* is better than or equal to *Adobe Photoshop Elements 4.0*. Several users report that labeling with our interactive tool is more like operating files in File Explorer, which is quite convenient. One of the users suggests we use more hot keys on key board to further speed up the operation. Several users are wondering whether *EasyAlbum* can be integrated into currently available softwares.

The photoshop system is a mature commercial software system with professional UI design, so it is interesting to see that our system as a test research demo can outperform the Photoshop system. This is mainly due to our innovative use of new technologies and the better design of user interaction, such as: "cluster annotation", "contextual re-ranking", and "ad hoc annotation".

CONCLUSION AND FUTURE WORK

We develop a system to efficiently label photos in family albums through simple and easy-to-use interactions. Face clustering labeling technique reduces workload of labeling people name in photos greatly. Contextual re-ranking provides an environment sensitive to user's interaction with the system, which boosts labeling productivity of the user. Ad hoc clustering is enabled to let users cluster and annotate freely when exploring and searching in the album, while progressively improving the performance of system at the same time.

We implement *EasyAlbum* based on the tags obtained through interactive labeling, and test our system through a carefully designed user study. The experiments verify the advantage of our system, especially when labeling large albums. The participants also respond positively to our system.

We are using MDS and other state-of-the-art visualization techniques to give users a more comprehensive impression of the similarities between the clusters. *Action-By-Stroke* interaction mechanisms will also be adopted to make handling the *FaceGroups* more easily and efficiently. We also plan to introduce more interesting user interaction scenarios, such as games, to the labeling process, to make the labeling process more enjoyable.

We will also make improvements to our system by adding convenient keyboard shortcuts and integrating some of our previous works of interactive photo segmentation and editing to the current system. This will make *EasyAlbum* more interesting to the users. Finally, we are planning to extend our work on other media file labeling and organization, such as musics and videos.

ACKNOWLEDGEMENTS
We thank Shu Liu, Qiong Yang, and Tie Liu for their help and contribution to the project.

REFERENCES
1. iView Media Pro. http://www.iview-multimedia.com.
2. ACDSee. http://www.acdsee.com.
3. Microsoft Diginal Image Suite. http://www.microsoft.com/products/imaging.
4. Picasa. http://picasa.google.com.
5. Photoshop Elements. http://www.adobe.com/products/photoshopelwin.
6. Flickr. http://www.flickr.com.
7. ESP Game. http://espgame.org.
8. Google Image Labeler. http://images.google.com/imagelabeler.
9. Riya. http://www.riya.com.
10. T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *Proc. the 8th European Conference on Computer Vision*, 2004.
11. F. Bach and M. Jordan. Learning spectral clustering. In *Proc. of Neural Info. Processing Systems*, 2003.
12. L. Chen, B. Hu, L. Zhang, M. Li, and H. Zhang. Face annotation for family photo album management. *International Journal of Image and Graphics (IJIG), Special Issue on Multimedia Data Storage and Management*, 3, 2003.
13. A. Girgensohn, J. Adcock, and L. Wilcox. Leveraging face recognition technology to find and organize photos. In *MIR '04: Proceedings of the 6th ACM SIGMM international workshop on Multimedia information retrieval*, pages 99–106, New York, NY, USA, 2004. ACM Press.
14. A. Graham, H. Garcia-Molina, A. Paepcke, and T. Winograd. Time as essence for photo browsing through personal digital libraries. In *Joint Conference on Digital Libraries*, pages 326–335, New York, NY, USA, 2002. ACM Press.
15. A. Kuchinsky, C. Pering, M. L. Creech, D. Freeze, B. Serra, and J. Gwizdka. Fotofile: a consumer multimedia organization and retrieval system. In *CHI '99: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 496–503, New York, NY, USA, 1999. ACM Press.
16. J. Platt. Autoalbum: Clustering digital photographs using probabalistic model merging, 2000.
17. J. C. Platt, M. Czerwinski, and B. A. Field. Phototoc: Automatic clustering for browsing personal photographs, 2002.
18. K. Rodden and K. R. Wood. How do people manage their digital photographs? In *CHI '03: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 409–416, New York, NY, USA, 2003. ACM Press.
19. B. Shneiderman and H. Kang. Direct annotation: A drag-and-drop strategy for labeling photos. In *Proc. of IEEE Conf. on Information Visualization*, 2000.
20. B. Suh and B. Bederson. Semi-automatic image annotation using event and torso identification. Technical report, Computer Science Department, University of Maryland, College Park, MD, 2004.
21. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, volume 1, pages 511–518, 2001.
22. L. von Ahn and L. Dabbish. Labeling images with a computer game. In *CHI '04: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326, New York, NY, USA, 2004. ACM Press.
23. X. Wang and X. Tang. A unified framework for subspace face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 26(9):1222–1228, 2004.
24. X. Wang and X. Tang. Random sampling for subspace face recognition. *Int. J. Comput. Vision*, 70(1):91–104, 2006.
25. L. Zhang, L. Chen, M. Li, and H. Zhang. Automated annotation of human faces in family albums. In *Proc. of ACM Multimedia*, 2003.
26. L. Zhang, Q. Yang, T. Bao, D. Vronay, and X. Tang. Imlooking: image-based face retrieval in online dating profile search. In *CHI '06: CHI '06 extended abstracts on Human factors in computing systems*, pages 1577–1582, New York, NY, USA, 2006. ACM Press.
27. W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Comput. Surv.*, 35(4):399–458, 2003.
28. Y. Zhou, L. Gu, and H.-J. Zhang. Bayesian tangent shape model: Estimating shape and pose parameters via bayesian inference. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, 2003.
29. Y. Zhou, W. Zhang, X. Tang, and H. Shum. A bayesian mixture model for multi-view face alignment. In *CVPR '05: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 2*, pages 741–746, Washington, DC, USA, 2005. IEEE Computer Society.